

Recursive Sparse Point Process Regression with Application to Spectrotemporal Receptive Field Plasticity Analysis

Alireza Sheikhattar, Jonathan B. Fritz, Shihab A. Shamma, and Behtash Babadi

Abstract—We consider the problem of estimating the sparse time-varying parameter vectors of a point process model in an online fashion, where the observations and inputs respectively consist of binary and continuous time series. We construct a novel objective function by incorporating a forgetting factor mechanism into the point process log-likelihood to enforce adaptivity and employ ℓ_1 -regularization to capture the sparsity. We provide a rigorous analysis of the maximizers of the objective function, which extends the guarantees of compressed sensing to our setting. We construct two recursive filters for online estimation of the parameter vectors based on proximal optimization techniques, as well as a novel filter for recursive computation of statistical confidence regions. Simulation studies reveal that our algorithms outperform several existing point process filters in terms of trackability, goodness-of-fit and mean square error. We finally apply our filtering algorithms to experimentally recorded spiking data from the ferret primary auditory cortex during attentive behavior in a click rate discrimination task. Our analysis provides new insights into the time-course of the spectrotemporal receptive field plasticity of the auditory neurons.

Index Terms—Adaptive filtering; point process models; compressed sensing; neural signal processing; receptive field plasticity.

I. INTRODUCTION

Analyses of spiking activity recorded from sensory neurons have revealed three main features: first, neuronal activity is stochastic and exhibits significant variability across trials; second, the spiking statistics often undergo rapid changes referred to as neuronal plasticity, in order to adapt to changing stimulus salience and behavioral context; and third, the tuning characteristics of sensory neurons to the stimuli exhibit a degree of sparsity. Examples include place cells in the hippocampus [1] and spectrotemporally tuned cells in the primary auditory cortex [2]. Hence, in order to gain insight into the functional mechanism of the underlying neural system, it is crucial to have a mathematical theory to simultaneously capture the stochasticity, dynamicity and sparsity of neuronal activity.

On one hand, the theory of point processes [3] has been recently adopted as a mathematical framework to model the stochasticity of neuronal data. Traditionally, these models have

been used to predict the likelihood of self-exciting processes such as earthquake occurrences [4], [5], but have recently found significant applications in the analysis of neuronal data [6]–[12].

On the other hand, classic results in signal processing such as the Least Mean Squares (LMS) and Recursive Least Squares (RLS) algorithms [13] have created a framework to efficiently capture the dynamics of the parameters in linear observation models. Existing solutions in computational neuroscience have adopted this framework to estimate the dynamics of neuronal activity. For instance, in [7] an LMS-type point process filter was introduced to study plasticity in hippocampal neurons. In [14], more general adaptive filtering solutions based on approximations to the Chapman-Kolmogorov equation were introduced. Although quite powerful in analyzing neuronal data, these solutions do not account for the sparsity of the underlying parameters.

Finally, the theory of compressed sensing (CS) has provided a novel methodology for measuring and estimating statistical models governed by sparse underlying parameters [15]–[17]. In particular, for *static* linear and generalized linear models (GLM) with random covariates and sparsity of the parameters, the CS theory characterizes sharp trade-offs between the number of measurement, sparsity, and estimation accuracy [16], [18]. The sparse solutions of CS are typically achieved using batch-mode convex programs and greedy techniques. In online settings, sparse adaptive filters have only been introduced in the context of linear systems governed by sparse parameters such as communication channels [19]–[21].

Despite significant progress in all these research fronts, a unified framework to simultaneously capture the stochasticity, dynamicity and sparsity of neuronal data is lacking. In this paper, we close this gap by integrating techniques from point process theory, adaptive filtering, and compressed sensing. To this end, we consider the problem of estimating time-varying stimulus modulation coefficients (e.g., receptive fields) from a sequence of binary observations in an online fashion. We model the spiking activity by a conditional Bernoulli point process, where the conditional intensity is a logistic function of the stimulus and its time lags. We will then design a novel objective function by incorporating the forgetting factor mechanism of RLS-type algorithms into the ℓ_1 -regularized maximum likelihood estimation of the point process parameters. We will present theoretical guarantees that extend those of CS theory and characterize fundamental trade-offs between the number of measurements, forgetting factor, model compressibility, and

A. Sheikhattar is with the Department of Electrical and Computer Engineering (ECE), University of Maryland, College Park, MD; J. B. Fritz is with the Institute for Systems Research (ISR), University of Maryland, College Park, MD; S. A. Shamma and B. Babadi are with ECE and ISR, University of Maryland, College Park, MD (e-mails: arsha89@umd.edu; ripple@umd.edu; sas@umd.edu; behtash@umd.edu).

Corresponding author: B. Babadi (e-mail: behtash@umd.edu).

estimation error of the underlying point processes in the non-asymptotic regime. We will next develop two adaptive filters for recursive estimation of the objective function based on proximal gradient techniques, as well as a filter for recursive computation of statistical confidence regions.

In order to validate our algorithms, we provide simulation studies which reveal that the proposed adaptive filtering algorithms significantly outperform existing point process filters in terms of goodness-of-fit, mean square error and trackability. We finally apply our proposed filters to multi-unit spiking data from ferret primary auditory cortex (A1) during passive stimulus presentation and during performance of a click rate discrimination task [22] in order to characterize the spectrotemporal receptive field (STRF) plasticity of A1 neurons. Application of our algorithm to these data provides new insights into the time course of attention-driven STRF plasticity, with over 3 orders of magnitude increase in temporal resolution from minutes to centiseconds, while capturing the underlying sparsity in a robust fashion. Aside from their theoretical significance, our results are particularly important in light of the recent technological advances in neural prostheses, which require real-time robust neuronal system identification from limited data.

The rest of the paper is organized as follows: In Section II, we present our notational conventions, preliminaries and problem formulation. In Section III, we introduce the main theoretical results of this paper, including the construction and stability analysis of the objective function, recursive filter development, and computation of confidence regions. Section IV provides numerical simulations as well as application to real data, followed by our concluding remarks in Section V. Technical details of Section III are presented in Appendices A–C.

II. PRELIMINARIES AND PROBLEM DEFINITION

We first give a brief introduction to point process models (see [3] for a detailed treatment). We will use the following notation throughout the paper. Parameter vectors are denoted by bold-face greek letters. For example, $\omega = [\omega_1, \omega_2, \dots, \omega_M]'$ denotes an M -dimensional parameter vector, with $[\cdot]'$ denoting the transpose operator.

Consider a stochastic process defined by a sequence of discrete events at random points in time, noted by $t_1^J = [t_1, t_2, \dots, t_J]'$, and a counting measure given by

$$dN(t) = \sum_{k=1}^J \delta(t - t_k), \quad \text{and} \quad N(t) = \int_0^t dN(u),$$

where $\delta(\cdot)$ is the Dirac's measure. The Conditional Intensity Function (CIF) for this process, denoted by $\lambda_t|H_t$, is defined as

$$\lambda(t|H_t) := \lim_{\varepsilon \rightarrow 0} \frac{\mathbb{P}(N(t+\varepsilon) - N(t) = 1|H_t)}{\varepsilon}, \quad (1)$$

where H_t denotes the history of the process as well as the covariates up to time t . The CIF can be interpreted as the *instantaneous rate* given the history of the process and the covariates. A point process with a CIF given by $\lambda(t|H_t)$ is defined as:

- 1) $N(0) = 0$
- 2) Given $0 = t_0 < t_1 < t_2 < \dots$, the random variables $N(t_k) - N(t_{k-1})$ are *conditionally* mutually independent.
- 3) For any $0 \leq t_1 < t_2$, $N(t_2) - N(t_1)$ is a Poisson random variable with probability distribution

$$\mathbb{P}(N(t_2) - N(t_1) = k) = \frac{\left(\int_{t_1}^{t_2} \lambda(t|H_t) dt \right)^k e^{-\int_{t_1}^{t_2} \lambda(t|H_t) dt}}{k!}.$$

A point process model is fully characterized by its CIF. For instance, $\lambda(t|H_t) = \lambda$ corresponds to the homogenous Poisson process with rate λ . A discretized version of this process can be obtained by binning $N(t)$ within an observation interval of $[0, T]$ by bins of length Δ , that is

$$n_t := N(t\Delta) - N((t-1)\Delta), \quad t = 1, 2, \dots, T \quad (2)$$

where $T := \lceil T/\Delta \rceil$. Throughout this paper, $\{n_t\}_{t=1}^T$ will be considered as the observed spiking sequence, which will be used for estimation purposes. Also, by approximating Eq. (1) for small $\Delta \ll 1$, and defining $\lambda_t := \lambda(t\Delta|H_{t\Delta})$, we have:

$$\begin{aligned} \mathbb{P}(n_t = 0) &= 1 - \lambda_t \Delta + o(\Delta), \\ \mathbb{P}(n_t = 1) &= \lambda_t \Delta + o(\Delta), \\ \mathbb{P}(n_t \geq 2) &= o(\Delta). \end{aligned} \quad (3)$$

In discrete time, the orderliness of the process is equivalent to the requirement that with high probability not more than one event fall into any given bin. In practice, this can always be achieved by choosing Δ small enough. An immediate consequence of Eq. (3) is that $\{n_t\}_{t=1}^T$ can be approximated by a sequence of Bernoulli random variables with success probabilities $\{\lambda_t \Delta\}_{t=1}^T$.

A popular class of models for the CIF is given by Generalized Linear Models (GLM). In its general form, a GLM consists of two main components: an observation model (which is given by (3) in this paper) and an equation expressing some (possibly nonlinear) function of the observation mean as a *linear* combination of the covariates. In neuronal systems, the covariates consist of extrinsic covariates (e.g., neural stimuli) as well as intrinsic covariates (e.g., the history of the process). In this paper, we only consider GLM models with purely extrinsic covariates, although most of our results can be generalized to incorporate intrinsic covariates as well.

Let s_t denote the stimulus at time bin t , $[\theta_0, \theta_1, \dots, \theta_{M-2}]'$ denote the vector of stimulus modulation parameters, and μ denote the baseline firing rate. We adopt a logistic regression model for the CIF as follows:

$$\text{logit}(\lambda_t \Delta) := \log \left(\frac{\lambda_t \Delta}{1 - \lambda_t \Delta} \right) = \mu + \sum_{i=0}^{M-2} \theta_i s_{t-i} \quad (4)$$

By defining $\omega := [\mu, \theta_0, \theta_1, \dots, \theta_{M-2}]'$ and $\mathbf{x}_t := [1, s_t, \dots, s_{t-M+2}]$, we can equivalently write:

$$\lambda_t \Delta = \text{logit}^{-1}(\omega' \mathbf{x}_t) := \frac{\exp(\omega' \mathbf{x}_t)}{1 + \exp(\omega' \mathbf{x}_t)} \quad (5)$$

The model above is also known as the logistic-link CIF model. Another popular model in the computational neuroscience literature is the log-link model where $\lambda_t \Delta = \exp(\omega' \mathbf{x}_t)$. The significance of the logistic-link model is that

$\text{logit}^{-1}(\cdot)$ maps the real line $(-\infty, +\infty)$ to the unit probability interval $(0, 1)$, making it a feasible model for describing statistics of binary events independent of the scaling of the covariates and modulation parameters.

Despite capturing the stimulus dependence in quite a general form, the GLM model in (5) represents a static model. We therefore generalize this model to the dynamic setting by allowing temporal variability of the modulation parameters:

$$\lambda_t \Delta = \text{logit}^{-1}(\omega'_t \mathbf{x}_t) = \frac{\exp(\omega'_t \mathbf{x}_t)}{1 + \exp(\omega'_t \mathbf{x}_t)} \quad (6)$$

where $\omega_t := [\mu_t, \theta_{t,0}, \theta_{t,1}, \dots, \theta_{t,M-2}]'$ represents the time-varying parameter vector at time t . Throughout the rest of the paper, we refer to \mathbf{x}_t and ω_t as the covariate vector and the modulation parameter vector at time t , respectively.

In our applications of interest, the modulation parameter vector ω exhibits a degree of sparsity [23], [24]. That is, only certain components in the stimulus modulation have significant contribution in determining the statistics of the process. These components can be thought of as the preferred or intrinsic tuning features of the underlying neuron. To be more precise, for a sparsity level $L < M$, we denote by $S \subset \{1, 2, \dots, M\}$ the support of the L highest elements of ω in absolute value, and by ω_L the best L -term approximation to ω . We also define

$$\sigma_L(\omega) := \|\omega - \omega_L\|_1 \quad (7)$$

to capture the compressibility of the parameter vector ω . Recall that for $\mathbf{x} \in \mathbb{R}^M$, the ℓ_1 -norm is defined as $\|\mathbf{x}\|_1 := \sum_{i=1}^M |x_i|$. When $\sigma_L(\omega) = 0$, the parameter ω is called L -sparse, and when $\sigma_L(\omega)$ is small compared to $\|\omega_L\|_1$, the parameter is called L -compressible [25].

Finally, the main estimation problem of this paper can be stated as follows: *given binary observations $\{n_t\}_{t=1}^T$ and covariates $\{\mathbf{x}_t\}_{t=-M+1}^T$ from a point process with a CIF given by Eq. (6), the goal is to estimate the M -dimensional L -compressible parameter vectors $\{\omega_t\}_{t=1}^T$ in an online and stable fashion.*

III. MAIN RESULTS

In this section, we will first describe the construction of an appropriate objective function for addressing our main estimation problem. We will then present a rigorous analysis of the maximizers of the objective function, which extends the results of CS to our setting. Next, we will introduce two adaptive filters to recursively maximize the objective function based on proximal gradient techniques. Finally, we will outline how statistical confidence bounds can also be constructed in a recursive fashion for our estimates.

A. ℓ_1 -regularized Exponentially Weighted Maximum Likelihood (ML)

Before proceeding with the construction of the objective function, we need to introduce more notational conventions. In order to have a framework allowing multi-timescale dynamics, we consider piece-wise constant dynamics for the parameter ω_t . That is, we assume that ω_t remains constant over windows of arbitrary length $W \geq 1$ samples, for some integer W . By segmenting the corresponding spiking data $\{n_t\}_{t=1}^T$ into $K :=$

$\frac{T}{W}$ windows of length W samples each, we assume that the CIF for each time point $(k-1)W+1 \leq t \leq kW$ is governed by $\omega_t = \omega_k$, for $k = 1, 2, \dots, K$. Note that number of spiking samples K is assumed to be an integer multiple of window size W , without loss of generality.

Invoking the Bernoulli approximation to the Poisson statistics for $\Delta \ll 1$, the log-likelihood of the observation n_t at time t can be expressed as:

$$\begin{aligned} \log p(n_t) &\approx n_t \log(\lambda_t \Delta) + (1 - n_t) \log(1 - \lambda_t \Delta) \\ &= n_t (\mathbf{x}'_t \omega_t) - \log(1 + \exp(\mathbf{x}'_t \omega_t)). \end{aligned} \quad (8)$$

Assuming conditional independence of the spiking events, the joint log-likelihood of the observations within window k is given by:

$$\begin{aligned} \mathcal{L}_k(\omega_k) &:= \sum_{j=1}^W n_{(k-1)W+j} \mathbf{x}'_{(k-1)W+j} \omega_k \\ &\quad - \log(1 + \exp(\mathbf{x}'_{(k-1)W+j} \omega_k)) \end{aligned} \quad (9)$$

In order to explicitly enforce adaptivity in the log-likelihood function, we adopt the forgetting factor mechanism of the RLS algorithm, where the log-likelihood of each window is exponentially weighted regressively in time, with a forgetting factor $0 < \beta \leq 1$. That is, the effective data log-likelihood up to and including window k is taken to be:

$$\mathcal{L}^\beta(\omega_k) := \sum_{i=1}^k \beta^{k-i} \mathcal{L}_i(\omega_k) \quad (10)$$

for some $0 < \beta \leq 1$. Note that for $\beta = 1$, $\mathcal{L}^1(\omega_k)$ coincides with the natural data log-likelihood. Moreover, if we replace the Bernoulli log-likelihood with the Gaussian log-likelihood, then $\mathcal{L}^\beta(\omega_k)$ coincides with the conventional RLS objective function.

Next, in order to explicitly enforce sparsity, we adopt the ℓ_1 -regularization mechanism of CS. That is, at window k , we seek an estimate of the form:

$$\hat{\omega}_k = \underset{\omega_k}{\operatorname{argmax}} \{ \mathcal{L}^\beta(\omega_k) - \gamma \|\omega_k\|_1 \} \quad (11)$$

where γ is a regularization parameter controlling the trade off between the log-likelihood fit and the sparsity of estimated parameters. Our theoretical analysis in the next subsection reveals appropriate choices for γ , β and the trade-offs therein.

B. Stability Analysis of the Objective Function

In order to quantify the trade-offs involving our choice of the objective function in Eq. (11), we proceed in the tradition of performance analysis result of the RLS algorithm [13] by characterizing the geometric properties of the estimates ω_k in a stationary environment where $\omega_k = \omega$ for all k . Our analysis, however, is quite general and avoids ad hoc assumptions such as direct averaging or covariate independence which are usually invoked in the analysis of least squares problems.

We need to make the following technical assumptions for our analysis:

1) The stimulus sequence $\{s_t\}_{t=-M+1}^T$ consists of independent (but not necessarily identically distributed) random variables with a variance of σ^2 which are uniformly

bounded by a constant $B > 0$ in absolute value. Note that with this assumption, two successive covariate vectors, say at times t and $t + 1$, given respectively by $\mathbf{x}_t = [1, s_{t-M+2}, s_{t-M+3}, s_{t-M+4}, \dots, s_t]$ and $\mathbf{x}_{t+1} = [1, s_{t-M+3}, s_{t-M+4}, \dots, s_t, s_{t+1}]$ are highly *dependent*, as they have $M - 3$ random variables in common. Hence, the independent assumption used in studying least squares problem is violated.

2) We further assume that for all times t , $0 < p_{\min} \leq \lambda_t \Delta \leq p_{\max} < 1$, for some constants p_{\min} and p_{\max} , i.e., the probability of spiking does not reach its extremal values of 0 and 1, but can get arbitrarily close. This assumption can be realized due to the boundedness of the covariates and appropriate normalization of the stimulus modulation coefficients, and does not result in any practical loss of generality.

We have the following theoretical result regarding the stability of the maximizers of the objective function:

Theorem 1: Suppose that binary observations from a point process with a CIF given by Eq. (6) are given over K windows of length W each. Consider a stationary environment with $\omega_k = \omega$ for all k and suppose that ω is L -compressible. Then, under assumptions (1) and (2), for a fixed positive constant $d > 0$, there exist constants C , C' , and C'' such that for $1 - \frac{C'}{L^2 \log M} \leq \beta < 1$, $K \geq \frac{\log 2}{\log(\frac{2}{\beta})}$ and a choice of $\gamma \geq C'' \sqrt{\frac{\log M}{1-\beta}}$, any solution $\hat{\omega}$ to (11) satisfies the bound

$$\|\hat{\omega} - \omega\|_2 \leq C \sqrt{(1-\beta)L \log M} + \sqrt{C \sigma_L(\omega)} \sqrt[4]{(1-\beta)L \log M},$$

with probability at least $1 - \frac{5}{M^d}$. The constants C , C' , and C'' are only functions of d , p_{\min} , p_{\max} , σ^2 , B , and W , and are explicitly given in Appendix A.

Proof: The proof is given in Appendix A. ■

Remarks. The result of Theorem 1 has four major implications. First, the error bound scales with \sqrt{L} , the sparsity level, as opposed to M , the ambient dimension of the parameter vector, which is consistent with results from CS, and results in the robustness of the estimate when the underlying parameter is sparse. Note that the bounds holds for general non-sparse ω , but is sharpest when $\sigma_L(\omega)$ is negligible, i.e., the parameter vector is nearly L -sparse.

Second, the theorem prescribes a lower bound on the forgetting factor akin to the bounds obtained in CS theory for the total number of observations. For instance, the result of [26] for CS under Toeplitz sensing measurements for the linear model requires $T = \mathcal{O}(L^2 \log M)$ number of measurements to achieve a similar scaling of the error bound. In our case, the role of the number of measurements is transferred to forgetting factor by taking $\frac{1}{1-\beta}$ as the *effective* length of the measurements. In the absence of the forgetting factor ($\beta = 1$), by a careful limiting process, our results require $T = \mathcal{O}(L^2 \log M)$ measurements. The latter case can be compared to the result of [18] for point process models with independent and identically distributed covariate vectors, which requires $\mathcal{O}(L \log M)$ for stability. The loss of $\mathcal{O}(L)$ is incurred due to the shift structure and hence high dependence of the covariate vectors in our case, as exemplified in assumption (1).

Third, the theorem reveals the scaling of the regularization parameter in terms of M and β . In particular, this scaling is significant as it reveals another role for the forgetting factor mechanism: not only the forgetting factor mechanism allows for adaptivity of the estimates, it also controls the scaling of the ℓ_1 -regularization term with respect to the log-likelihood term. Fourth, unlike conventional results in the analysis of adaptive filters which concern the expectation of the error in the asymptotic regime, our result holds for a single realization with probability polynomially approaching 1, in the non-asymptotic regime.

Note that the objective function is clearly concave, and assuming that the matrix of the covariate vectors is full-rank, will be strictly concave with a unique maximizer. However, the result of Theorem 1 does not require the uniqueness of the maximizer and holds for any maximizer of the objective function. In the next section, we will proceed with the development of recursive filters to track the maximizer of the objective function in the more general time-varying setting.

C. Algorithm Development

Several standard optimization techniques, such as interior point methods, can be used to find the maximizer of (11). However, most of these techniques operate in batch mode and do not meet the real-time requirements of the adaptive filtering setting where the observations arrive in a streaming fashion. In order to avoid the increasing runtime complexity and memory requirements of the batch-mode computation, we seek a recursive approach which can perform low-complexity updates in an online fashion upon the arrival of new data in order to form the estimates. To this end, we adopt the proximal gradient approach. A version of the proximal gradient algorithm is given in Appendix B. Each iteration of the algorithm moves the previous iterate along the gradient of the log-likelihood function, which will then pass through a shrinkage operator.

Before describing further details, we introduce a more compact notation for convenience. Let $\mathbf{n}_k := [n_{(k-1)W+1}, n_{(k-1)W+2}, \dots, n_{kW}]'$ denote the vector of observed spikes within window k , for $k = 1, 2, \dots, K$. Similarly, let $\lambda_k := [\lambda_{(k-1)W+1}, \lambda_{(k-1)W+2}, \dots, \lambda_{kW}]'$ denote the vector of CIFs within window k . By extending the domain of the $\text{logit}^{-1}(\cdot)$ to vectors in a component-wise fashion, we can express $\lambda_k \Delta$ as:

$$\lambda_k \Delta = \text{logit}^{-1}(\mathbf{X}_k \omega_k) \quad (12)$$

where $\mathbf{X}_k := [\mathbf{x}_{(k-1)W+1}, \mathbf{x}_{(k-1)W+2}, \dots, \mathbf{x}_{kW}]'$ is the data matrix of size $W \times M$ with rows corresponding to the covariate vectors in window k . Suppose that at window k , we have an iterate denoted by $\hat{\omega}_k^{(\ell)}$, for $\ell = 0, 1, \dots, R$, with R being an integer denoting the total number of iterations. The gradient of $\mathcal{L}^\beta(\cdot)$ evaluated at $\hat{\omega}_k^{(\ell)}$ can be written as:

$$\nabla_{\omega} \mathcal{L}^\beta(\hat{\omega}_k^{(\ell)}) = \sum_{i=1}^k \beta^{k-i} \mathbf{X}_i' \varepsilon_i(\hat{\omega}_k^{(\ell)}) =: \mathbf{g}_k(\hat{\omega}_k^{(\ell)}) \quad (13)$$

where $\varepsilon_i(\cdot) := \mathbf{n}_i - \lambda_i(\cdot) \Delta$ represents the innovation vector of the point process at window i . The innovation vector ε_i can be thought of as the counterpart of the conventional innovation

vector in adaptive filtering of linear models. The proximal gradient iteration for the ℓ_1 -regularization can be written in the compact form as:

$$\hat{\omega}_k^{(\ell+1)} = \mathcal{S}_{\gamma\alpha} \left(\hat{\omega}_k^{(\ell)} + \alpha \mathbf{g}_k \left(\hat{\omega}_k^{(\ell)} \right) \right) \quad (14)$$

where $\mathcal{S}_\tau(\cdot)$ is the element-wise soft thresholding operator at a level of τ given in Appendix B. The final estimate at window k is obtained following the R th iteration, and is denoted by $\hat{\omega}_k := \hat{\omega}_k^{(R)}$. In order to achieve a recursive updating rule for \mathbf{g}_k , we can rewrite Eq. (13) as:

$$\mathbf{g}_k \left(\hat{\omega}_k^{(\ell)} \right) = \beta \mathbf{g}_{k-1} \left(\hat{\omega}_k^{(\ell)} \right) + \mathbf{X}'_k \varepsilon_k \left(\hat{\omega}_k^{(\ell)} \right) \quad (15)$$

However, in an adaptive setting, we only have access to values of \mathbf{g}_{k-1} evaluated at $\hat{\omega}_{k-1}^{(1:L)}(\cdot)$! In order to turn Eq. (15) into a fully recursive updating rule, all the previous CIF vectors $\{\lambda_i(\cdot)\}_{i=1}^{k-1}$ should be recalculated at the most recent set of iterates $\hat{\omega}_k^{(1:L)}(\cdot)$. In order to overcome this computational burden, we exploit the smoothness of the logistic function and employ the Taylor series expansion of the CIF to approximate the required recursive update. In what follows, we consider the zeroth order and first order expansions, which result in two distinct, yet fully recursive, updating rules for Eq. (15).

Zeroth Order Expansion: By retaining only the first term in the Taylor series expansion of the CIF $\lambda_i \left(\hat{\omega}_k^{(\ell)} \right)$ around $\hat{\omega}_i$, we get:

$$\lambda_i \left(\hat{\omega}_k^{(\ell)} \right) \Delta \approx \lambda_i \left(\hat{\omega}_i \right) \Delta \quad (16)$$

where $\lambda_i \Delta = \text{logit}^{-1}(\mathbf{X}_i \hat{\omega}_i)$. Substituting this approximation in Eq. (13), we can express the zeroth order approximation to the gradient at window k , denoted by $\mathbf{g}_k^0(\cdot)$, as:

$$\mathbf{g}_k^0 \left(\hat{\omega}_k^{(\ell)} \right) = \sum_{i=1}^k \beta^{k-i} \mathbf{X}'_i \varepsilon_i \left(\hat{\omega}_i \right) \quad (17)$$

It is then straightforward to obtain a recursive form as:

$$\mathbf{g}_k^0 \left(\hat{\omega}_k^{(\ell)} \right) = \beta \mathbf{g}_{k-1}^0 \left(\hat{\omega}_k^{(\ell)} \right) + \mathbf{X}'_k \varepsilon_k \left(\hat{\omega}_k^{(\ell)} \right) \quad (18)$$

The shrinkage step will be then given by:

$$\hat{\omega}_k^{(\ell+1)} = \mathcal{S}_{\gamma\alpha} \left(\hat{\omega}_k^{(\ell)} + \alpha \mathbf{g}_k^0 \left(\hat{\omega}_k^{(\ell)} \right) \right) \quad (19)$$

We refer to the resulting filter as the ℓ_1 -regularized Point Process Filter of the Zeroth Order (ℓ_1 -PPF₀). A pseudo-code is given in Algorithm 1.

Algorithm 1 ℓ_1 -regularized Point Process Filter of the Zeroth Order (ℓ_1 -PPF₀)

Inputs: $\mathbf{n}_k, \mathbf{X}_k, \mathbf{g}_{k-1}, \hat{\omega}_k^{(0)}$, and R .

```

1: for  $\ell = 0, \dots, R-1$  do
2:    $\lambda_k \Delta = \text{logit}^{-1} \left( \mathbf{X}_k \hat{\omega}_k^{(\ell)} \right)$ 
3:    $\varepsilon_k = \mathbf{n}_k - \lambda_k \Delta$ 
4:    $\mathbf{g}_k = \beta \mathbf{g}_{k-1} + \mathbf{X}'_k \varepsilon_k$ 
5:    $\hat{\omega}_k^{(\ell+1)} = \mathcal{S}_{\gamma\alpha} \left[ \hat{\omega}_k^{(\ell)} + \alpha \mathbf{g}_k \right]$ 
6: end for
```

Output: $\hat{\omega}_k := \hat{\omega}_k^{(R)}$.

First Order Expansion: If instead, we retain the first two terms in the Taylor expansion, Eq. (16) will be replaced by:

$$\lambda_i \left(\hat{\omega}_k^{(\ell)} \right) \Delta \approx \lambda_i \left(\hat{\omega}_i \right) \Delta + \Lambda_i \left(\hat{\omega}_i \right) \mathbf{X}_i \left(\hat{\omega}_k^{(\ell)} - \hat{\omega}_i \right) \quad (20)$$

where $\Lambda_i(\hat{\omega}_i)$ is a diagonal $W \times W$ matrix with the (m, m) th diagonal element given by $\lambda_{(i-1)W+m} \Delta (1 - \lambda_{(i-1)W+m} \Delta)$. Using the first order approximation above, we can improve the resulting approximation to the gradient, denoted by \mathbf{g}_k^1 , as:

$$\mathbf{g}_k^1 \left(\hat{\omega}_k^{(\ell)} \right) = \sum_{i=1}^k \beta^{k-i} \mathbf{X}'_i \left(\varepsilon_i \left(\hat{\omega}_i \right) - \Lambda_i \left(\hat{\omega}_i \right) \mathbf{X}_i \left(\hat{\omega}_k^{(\ell)} - \hat{\omega}_i \right) \right)$$

By defining:

$$\mathbf{u}_k^{(\ell)} := \sum_{i=1}^k \beta^{k-i} \mathbf{X}'_i \left(\varepsilon_i \left(\hat{\omega}_i \right) + \Lambda_i \left(\hat{\omega}_i \right) \mathbf{X}_i \hat{\omega}_i \right) \quad (21)$$

$$\mathbf{B}_k^{(\ell)} := \sum_{i=1}^k \beta^{k-i} \mathbf{X}'_i \Lambda_i \left(\hat{\omega}_i \right) \mathbf{X}_i, \quad (22)$$

we can express $\mathbf{g}_k^1 \left(\hat{\omega}_k^{(\ell)} \right)$ as:

$$\mathbf{g}_k^1 \left(\hat{\omega}_k^{(\ell)} \right) = \mathbf{u}_k^{(\ell)} - \mathbf{B}_k^{(\ell)} \hat{\omega}_k^{(\ell)}. \quad (23)$$

It is then straightforward to check that both \mathbf{u}_k and \mathbf{B}_k can be updated recursively [19] as:

$$\mathbf{u}_k^{(\ell)} = \beta \mathbf{u}_{k-1}^{(R)} + \mathbf{X}'_k \left(\varepsilon_k \left(\hat{\omega}_k^{(\ell)} \right) + \Lambda_k \left(\hat{\omega}_k^{(\ell)} \right) \mathbf{X}_k \hat{\omega}_k^{(\ell)} \right)$$

$$\mathbf{B}_k^{(\ell)} = \beta \mathbf{B}_{k-1}^{(R)} + \mathbf{X}'_k \Lambda_k \left(\hat{\omega}_k^{(\ell)} \right) \mathbf{X}_k$$

Note that the update rules for both $\mathbf{B}_k^{(\ell)}$ and $\mathbf{u}_k^{(\ell)}$ involve simple rank- W operations. The shrinkage step is then given by:

$$\hat{\omega}_k^{(\ell+1)} = \mathcal{S}_{\gamma\alpha} \left(\left(\mathbf{I} - \alpha \mathbf{B}_k^{(\ell)} \right) \hat{\omega}_k^{(\ell)} + \alpha \mathbf{u}_k^{(\ell)} \right) \quad (24)$$

We refer to the resulting filter as the ℓ_1 -regularized Point Process Filter of the First Order (ℓ_1 -PPF₁). A pseudo-code is given in Algorithm 2.

Remark. The computational complexity of ℓ_1 -PPF₀ and ℓ_1 -PPF₁ algorithms can be shown to be linear and quadratic in M per iteration, respectively. Our results in Section IV will reveal that both filters outperform existing filters of the same complexity, respectively. Furthermore, ℓ_1 -PPF₁ exhibits superior performance over ℓ_1 -PPF₀ as expected, although with a cost of $\mathcal{O}(M)$ in computational complexity per iteration.

Algorithm 2 ℓ_1 -regularized Point Process Filter of the First Order (ℓ_1 -PPF₁)

Inputs: $\mathbf{n}_k, \mathbf{X}_k, \mathbf{u}_{k-1}, \mathbf{B}_{k-1}, \hat{\omega}_k^{(0)}$, and R .

```

1: for  $\ell = 0, \dots, R-1$  do
2:    $\lambda_k \Delta = \text{logit}^{-1} \left( \mathbf{X}_k \hat{\omega}_k^{(\ell)} \right)$ 
3:    $\varepsilon_k = \mathbf{n}_k - \lambda_k \Delta$ 
4:    $(\Lambda_k)_{m,m} = (\lambda_k)_{m,m} \Delta (1 - (\lambda_k)_{m,m} \Delta), m = 1, \dots, W$ 
5:    $\mathbf{u}_k = \beta \mathbf{u}_{k-1} + \mathbf{X}'_k \left( \varepsilon_k + \Lambda_k \mathbf{X}_k \hat{\omega}_k^{(\ell)} \right)$ 
6:    $\mathbf{B}_k = \beta \mathbf{B}_{k-1} + \mathbf{X}'_k \Lambda_k \mathbf{X}_k$ 
7:    $\mathbf{g}_k = \mathbf{u}_k - \mathbf{B}_k \hat{\omega}_k^{(\ell)}$ 
8:    $\hat{\omega}_k^{(\ell+1)} = \mathcal{S}_{\gamma\alpha} \left[ \hat{\omega}_k^{(\ell)} + \alpha \mathbf{g}_k \right]$ 
9: end for
```

Output: $\hat{\omega}_k := \hat{\omega}_k^{(R)}$.

D. Constructing Confidence Intervals

Characterizing the statistical confidence bounds for the estimates is of utmost importance in neural data analysis, as it allows to test the validity of various hypotheses. Although construction of confidence bounds for linear models in the absence of regularization is well understood and widely applied, regularized ML estimates are usually deemed as point estimates for which the construction of statistical confidence regions is not straightforward. A series of remarkable results in high-dimensional statistics [27]–[29] have recently addressed this issue by providing techniques to construct confidence intervals for ℓ_1 -regularized ML estimates of linear models as well as GLMs. These approaches are based on a careful inspection of the Karush-Kuhn-Tucker (KKT) conditions for the regularized estimates, which admits a procedure to decompose the estimates into a bias term plus an asymptotically Gaussian term (referred to as ‘de-sparsifying’ in [28]), which can be computed using a nodewise regression [30] of the covariates.

In what follows, we give a brief description of how the methods of [28] apply to our setting, and leave the details to Appendix C. Using the result of [28], the estimate ω_k as the maximizer of (11) can be decomposed as:

$$\hat{\omega}_k = \hat{\Theta}_k \mathbf{g}_k(\hat{\omega}_k) + \hat{\mathbf{w}}_k \quad (25)$$

where $\hat{\Theta}_k$ is an approximate inverse to the Hessian of $\mathcal{L}^\beta(\omega)$ evaluated at $\hat{\omega}_k$, \mathbf{g}_k is the gradient of $\mathcal{L}^\beta(\omega)$ previously defined in Eq. (13), and $\hat{\mathbf{w}}_k$ is an unbiased and asymptotically Gaussian random vector with a covariance matrix of $\text{cov}(\hat{\mathbf{w}}_k) = \hat{\Theta}_k \mathbf{G}_k(\hat{\omega}_k) \hat{\Theta}_k'$, with

$$\mathbf{G}_k(\hat{\omega}_k) := \sum_{i=1}^k \beta^{2(k-i)} \mathbf{X}_i' \boldsymbol{\varepsilon}_i(\hat{\omega}_k) \boldsymbol{\varepsilon}_i(\hat{\omega}_k)' \mathbf{X}_i. \quad (26)$$

The first term in Eq. (25) is a bias term which can be directly computed given $\hat{\Theta}_k$. Given $\text{cov}(\hat{\mathbf{w}}_k)$, statistical confidence bounds for the second term at desired levels can be constructed

Algorithm 3 Recursive Construction of the Confidence Regions for the m th Component of $\hat{\mathbf{w}}_k$.^a

Inputs: \mathbf{n}_k , \mathbf{X}_k , \mathbf{u}_k , \mathbf{B}_k , and $\hat{\omega}_k$, \mathbf{G}_{k-1} , m , γ_m , and $\hat{\psi}_m^{(0)}$.

- 1: $\mathbf{g}_k = \mathbf{u}_k - \mathbf{B}_k \hat{\omega}_k$
- 2: $\mathbf{G}_k = \beta^2 \mathbf{G}_{k-1} + \mathbf{X}_k' \boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k' \mathbf{X}_k$
- 3: **for** $\ell = 0, \dots, R-1$ **do**
- 4: $\hat{\psi}_m^{(\ell+1)} = \mathcal{S}_{\gamma_m \alpha} \left[\hat{\psi}_m^{(\ell)} - \alpha \left((\mathbf{B}_k)_{m, \setminus m} - (\mathbf{B}_k)_{\setminus m, m} \hat{\psi}_m^{(\ell)} \right) \right]$
- 5: **end for**
- 6: $\tau_m^2 = (\mathbf{B}_k)_{m, m} - \hat{\psi}_m^{(R)} (\mathbf{B}_k)_{m, \setminus m}'$
- 7: $(\mathbf{c})_m = 1$, and $(\mathbf{c})_{\setminus m} = -\hat{\psi}_m^{(R)}$
- 8: $(\hat{\Theta}_k)_m = \frac{1}{\tau_m^2} \mathbf{c}$
- 9: $\hat{\sigma}_{k, m}^2 := (\hat{\Theta}_k)_m \mathbf{G}_k(\hat{\omega}_k) (\hat{\Theta}_k)_m'$
- 10: $(\hat{\mathbf{w}}_k)_m = (\hat{\omega}_k)_m - (\hat{\Theta}_k)_m \mathbf{g}_k$

Output: $\mathcal{CR}_{k, m} := [(\hat{\mathbf{w}}_k)_m \pm \Phi^{-1}(1 - \alpha/2) \hat{\sigma}_{k, m}]$

^aFor a matrix $\mathbf{A} \in \mathbb{R}^{M \times M}$, we denote by $(\mathbf{A})_{m, \setminus m}$ the m th row with the m th element removed, and by $(\mathbf{A})_{\setminus m, m}$ the submatrix of \mathbf{A} with both the m th row and column removed.

in a standard way. The main technical issue in the aforementioned procedure in our setting is the computation of $\hat{\Theta}_k$ in a recursive fashion. Since the rows of $\hat{\Theta}_k$ are computed using ℓ_1 -regularized least squares, we use the SPARLS algorithm [19] as an efficient method to carry out the computation in a recursive fashion. Algorithm 3 summarized the recursive computation of confidence intervals for the m th component of $\hat{\mathbf{w}}_k$.

IV. APPLICATIONS

In this section, we will apply the proposed algorithms to simulated data as well as real spiking data from the ferret primary auditory cortex. In our simulation studies, we compare the performance of our proposed filters with two of the state-of-the-art point process filters, namely the steepest descent point process filter (SDPPF) [7] and the stochastic state point process filter (SSPPF) [14]. These adaptive filters are based on approximate solutions to the Chapman-Kolmogorov forward equation obtained by a steepest descent and a Gaussian approximation procedure, respectively.

A. Simulation Study 1: MSE and Sparse Recovery Learning Curves

First, we consider a stationary environment where ω is constant over time. We use a bin size of $\Delta = 1$ ms and window size of $W = 1$ sample, for a total observation window of $\mathcal{T} = 30$ sec ($K = 30000$). The length of the parameter vector $\omega = [\mu, \theta]$ is chosen as $M = 101$. For each realization, we draw a sparse parameter vector θ of fixed length $M - 1 = 100$ and sparsity $L = 3$. The support S and values of the nonzero components of θ are chosen randomly and the values are normalized so that $\|\theta\|_2 = 10$. The stimulus input sequence $\{s_k\}_{k=-M+1}^K$ is drawn from an i.i.d. Gaussian distribution $\mathcal{N}(0, \sigma^2)$. The binary spike train $\{n_k\}_{k=1}^K$ is generated as a single realization of conditionally independent Bernoulli trials with success rate $\lambda_k \Delta$. The stimulus variance σ^2 is chosen as $\sigma^2 = 0.01$ small enough so that the average spiking rate $\lambda_{\text{avg}} \Delta = 0.13 \ll 1$ to ensure that the Bernoulli approximation is valid. All the simulations are done with $R = 1$ iteration per time step. The step size α is chosen as $\alpha \simeq 9 \times 10^{-4}$ (See Appendix B for details).

For a given pair of (α, β) parameters, we select an optimal value for the regularization parameter γ by performing a two-fold even-odd cross validation procedure: first, the data are split into two sets of even and odd samples in an interleaved manner. Then, one set is used as the training set for estimation of the parameter vector ω_k and the other is used to assess the goodness-of-fit of the estimates $\hat{\omega}_k$ with respect to the log-likelihood of the observations. We repeat the process switching the role of the two sets and take the average as the overall measure of fit.

Let $\hat{\mathbb{E}}$ denote the averaging operator with respect to realizations. We consider two performance metrics: the normalized mean squared error (MSE) defined as $\text{MSE}_k := \hat{\mathbb{E}} \|\hat{\omega}_k - \omega_k\|^2 / \hat{\mathbb{E}} \|\omega_k\|^2$ to evaluate MSE performance at time step k ; and the out-of-support energy defined as $\text{SPM}_k := \hat{\mathbb{E}} \|\hat{\theta}_k - (\hat{\theta}_k)_S\|^2 / \hat{\mathbb{E}} \|\hat{\theta}_k\|^2$ to represent a sparsity metric (SPM). Ideally, SPM_k must be equal to zero at all times. The averaging is carried out over a sufficiently large number of runs.

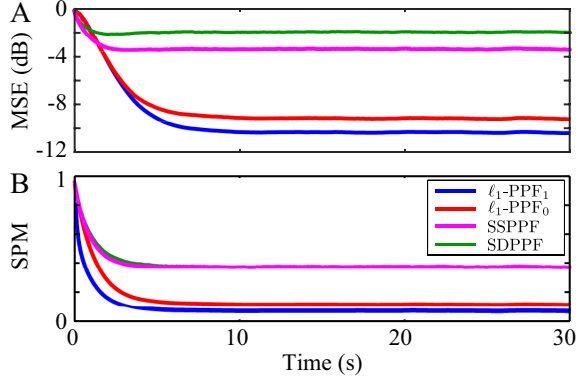


Fig. 1. Learning curves of the adaptive filtering algorithms in a stationary environment. A) MSE vs. time, B) SPM vs. time.

Figure 1 shows the corresponding learning curves for the four algorithms. According to Figure 1-A, the ℓ_1 -PPF₁ achieves the lowest stationary MSE measure of -10.4 dB, followed ℓ_1 -PPF₀ which achieves an MSE of -9.2 dB. The SSPPF and SDPPF algorithms respectively achieve an MSE of -3.35 dB and -1.9 dB, which reveals a gap of ≈ 7 dB with respect to our proposed filters. Note that this result is consistent with the prediction of Theorem 1 and highlights the MSE gain achieved by ℓ_1 -regularization, as opposed to ML, when the underlying parameter is sparse.

B. Simulation Study 2: Tracking and Goodness-of-fit Performance

In the second simulation scenario, we consider a more realistic setting where ω_k evolves in time. Furthermore, as in the case of real data applications, we assume that the support of ω_k is not available as a performance benchmark and resort to statistical goodness-of-fit test. These tests for point process models have been developed as an application of the time-rescaling theorem [31], [32] and consist of the Kolmogorov-Smirnov (KS) test for assessing the conditional intensity estimation accuracy, and the Autocovariance Function (ACF) test to assess the conditional independence assumption. We skip the details in the interest of space, and refer the readers to the aforementioned references for a detailed treatment.

As in the previous case, we consider a bin size of $\Delta = 1$ ms, window size of $W = 1$, and a total observation window of $\mathcal{T} = 60$ sec ($K = 60000$ bins). The stimulus is generated as in the previous case. For the parameter vector ω_k , we choose a fixed baseline rate of $\mu_k = -2.51$ to set the baseline spiking rate to $\lambda_{avg}\Delta \approx 0.1$, and select a sparse modulation vector θ_k of length $M = 100$ with a support $S = \{1, 10, 20\}$ of size $L = 3$, and respective values of $(\theta_k)_{\{1,10,20\}} = \{10, -5, 5\}$ for $k \leq K/2$. Halfway through the test, at $k = K/2 + 1$, the largest component $(\theta_k)_1$, drops rapidly and linearly to 0, within a window of length 1 sec and remains zero for the rest of the run.

Figure 2 shows the performance of all four algorithms in the aforementioned setting. Each row (A through D) shows the true time-varying parameter vector (dashed traces) as well as the filtered estimates (solid traces) in the left panel. In particular, the gray solid traces show the out-of-support components which must ideally be equal to zero. The colored hulls around $(\hat{\theta}_k)_1$ show the 95% confidence intervals (note that confidence

intervals for SDPPF cannot be directly obtained and require averaging over multiple realizations). The middle and right panels show the KS and ACF test results at a 95% confidence, respectively. For the quadratic algorithms ℓ_1 -PPF₁ and SSPPF, a forgetting factor of $\beta = 0.9995$ is chosen. The regularization parameter for ℓ_1 -PPF₁ is chosen as $\gamma = 0.5$, obtained by the aforementioned two-fold even-odd cross validation. For the first order algorithm ℓ_1 -PPF₀, a smaller forgetting factor of $\beta = 0.995$ is chosen to ensure stability, and a value of $\gamma = 0.1$ is used based on cross validation. These settings ensure that all the algorithms are tuned in their optimal operating point for fairness of comparison.

Figure 2-A and 2-B reveal three striking performance gaps between the two second-order algorithms (with the same computational complexity, quadratic in M): first, the out-of-support components (gray traces) of ℓ_1 -PPF₁ are significantly smaller than those of SSPPF; second, the confidence regions of ℓ_1 -PPF₁ are narrower than those of SSPPF; and third, ℓ_1 -PPF₁ fully passes the KS test, while SSPPF marginally does so. Similarly, comparing the two first order algorithms (with the same computational complexity, linear in M) Figure 2-C and 2-D reveal that the ℓ_1 -PPF₀ significantly suppresses the out-of-support components as compared to SDPPF. Moreover, ℓ_1 -PPF₀ provides confidence bounds, which cannot be directly obtained for SDPPF. Finally, ℓ_1 -PPF₀ marginally fails the KS test, whereas SDPPF does so significantly. Both algorithms fail the ACF test, which shows that the second-order corrections embedded in ℓ_1 -PPF₁ and SSPPF is necessary to achieve a better goodness-of-fit, which a price of higher computational complexity.

We also inspect the estimated firing probability $\lambda_k(\hat{\omega}_k)\Delta$ for the four algorithms in Figure 3. In addition, we include the probability estimated by the normalized reverse correlation (NRC) method, which is commonly used in neural data analysis, and fits the modulation parameters using a linear model. Figure 3 shows the true spiking probability (blue solid trace) and the resulting spikes (black vertical lines). In the subsequent rows (B through F), the true and estimated probabilities are shown by dashed blue and solid red traces, respectively. A comparison of all the rows reveals that ℓ_1 -PPF₁ and ℓ_1 -PPF₀ outperform SSPPF and SDPPF, respectively, in terms of estimating the true probability. The NRC method is inferior to the preceding four algorithms, and results in negative estimates of the probability due to its use of a linear model (as opposed to logistic).

C. Application to Real Data: Dynamic Analysis of Spectrotemporal Receptive Field Plasticity

The responses of neurons in the primary auditory cortex (A1) can be characterized by their spectrotemporal receptive fields (STRFs), where each neuron is tuned to a specific region in the time-frequency plane, and only significantly spikes when the acoustic stimulus contains spectrotemporal contents matching its tuning region [2] (See Figure 4, top row, leftmost panel). Several experimental studies have revealed that receptive fields undergo rapid changes in their characteristics during attentive behavior in order to capture salient stimulus modulations [22], [33], [34]. In [22], it is suggested that this rapid plasticity

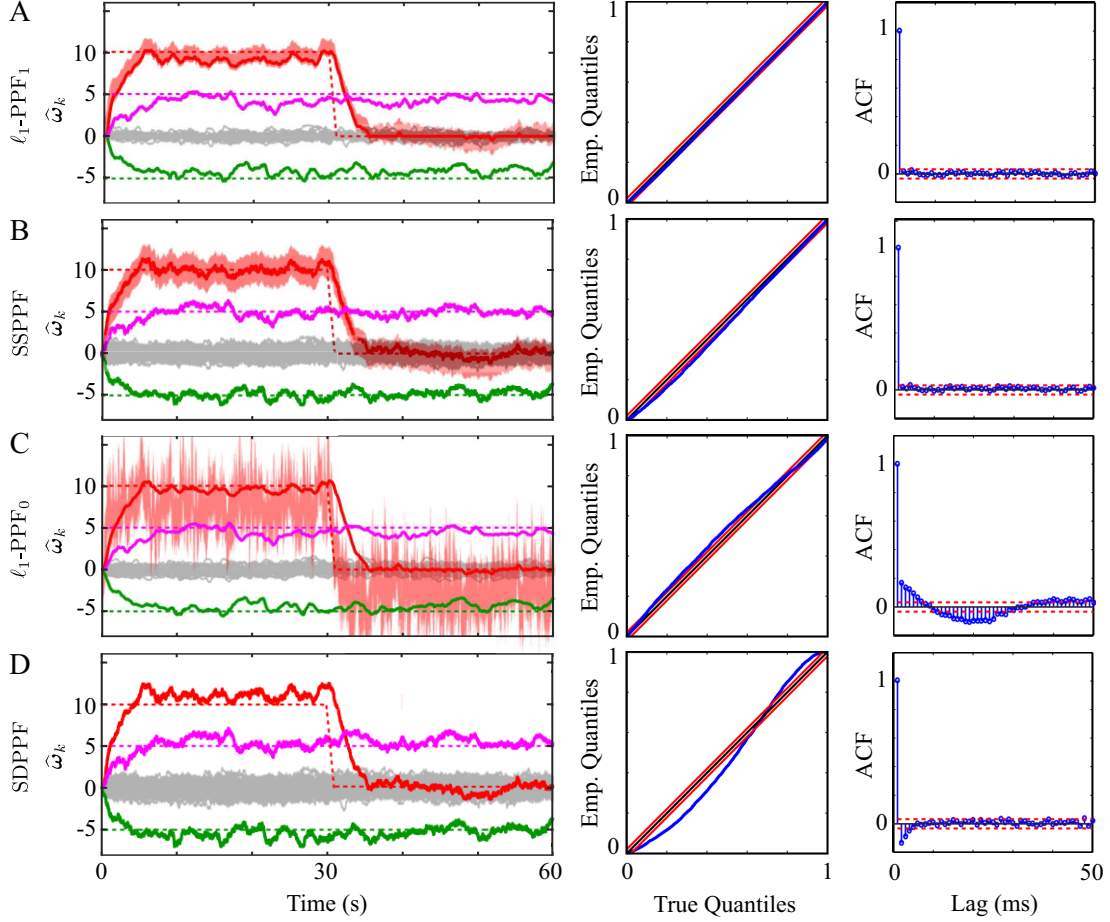


Fig. 2. Performance comparison of the adaptive filtering algorithms: A) ℓ_1 -PPF₁, B) SSPPF, C) ℓ_1 -PPF₀, and D) SDPPF. In each row, the left panel shows the true parameter vector with dashed traces and the estimates with solid traces. Colored hulls show the 95% confidence intervals for one of the components. The middle and right panels show the corresponding KS and ACF test plots, respectively. Red traces show confidence regions at a level of 95% for both tests.

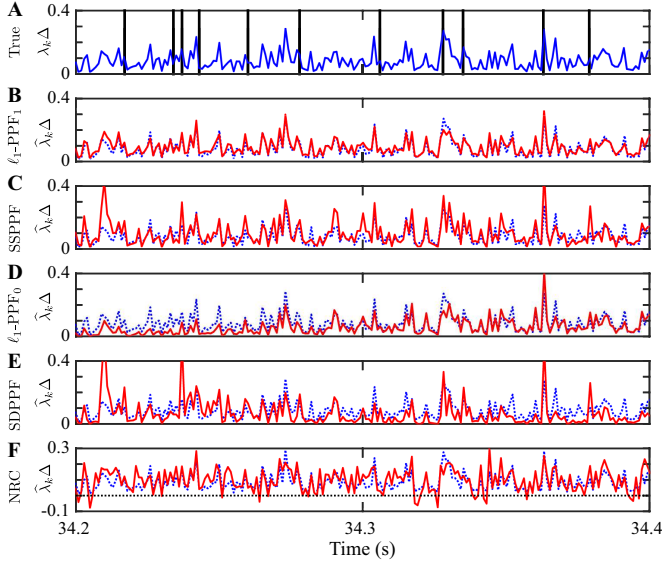


Fig. 3. Firing rate estimates for adaptive filtering algorithms within an interval of $34.2 \text{ s} \leq t \leq 34.4 \text{ s}$: A) true rate (blue solid trace) and spikes (black vertical lines), B) ℓ_1 -PPF₁, C) SSPPF, D) ℓ_1 -PPF₀, E) SDPPF, and F) normalized reverse correlation (NRC). In rows B through F, the dashed blue traces and solid red traces show the true rate and the estimated rate, respectively.

has a significant role in the functional processes underlying active listening. However, most of the widely-used estimation techniques (e.g., normalized reverse correlation) provide static estimates of the receptive field with a temporal resolution of the order of minutes. Moreover, they do not systematically capture the inherent sparsity manifested in the receptive field characteristics.

In the context of our model, the STRF can be modeled as an $(I \times J)$ -dimensional matrix, where I and J denote the number of time lags and frequency bands, respectively. By vectorizing this matrix, we obtain an $(M - 1)$ -dimensional vector θ_k at window k , where $M = I \times J + 1$. Augmenting the baseline rate parameter μ_k , we can model the activity of the A1 neurons using the logistic CIF with a parameter $\omega_k := [\mu_k, \theta_k]'$. The stimulus vector at time t , s_t is given by the vectorized version of the spectrogram of the acoustic stimulus with J frequency bands and I lags. In order to capture the sparsity of the STRF in the time-frequency plane, we further represent θ_k over a Gabor time-frequency dictionary consisting of Gaussian windows centered around a regular subset of the $I \times J$ time-frequency plane. That is, for $\theta_k = \mathbf{F}\xi_k$, where \mathbf{F} is the dictionary matrix and ξ_k is the sparse representation of the STRF. The estimation procedures of this paper can be applied to ξ_k , by absorbing the dictionary matrix into the data matrix \mathbf{X}_k at window k .

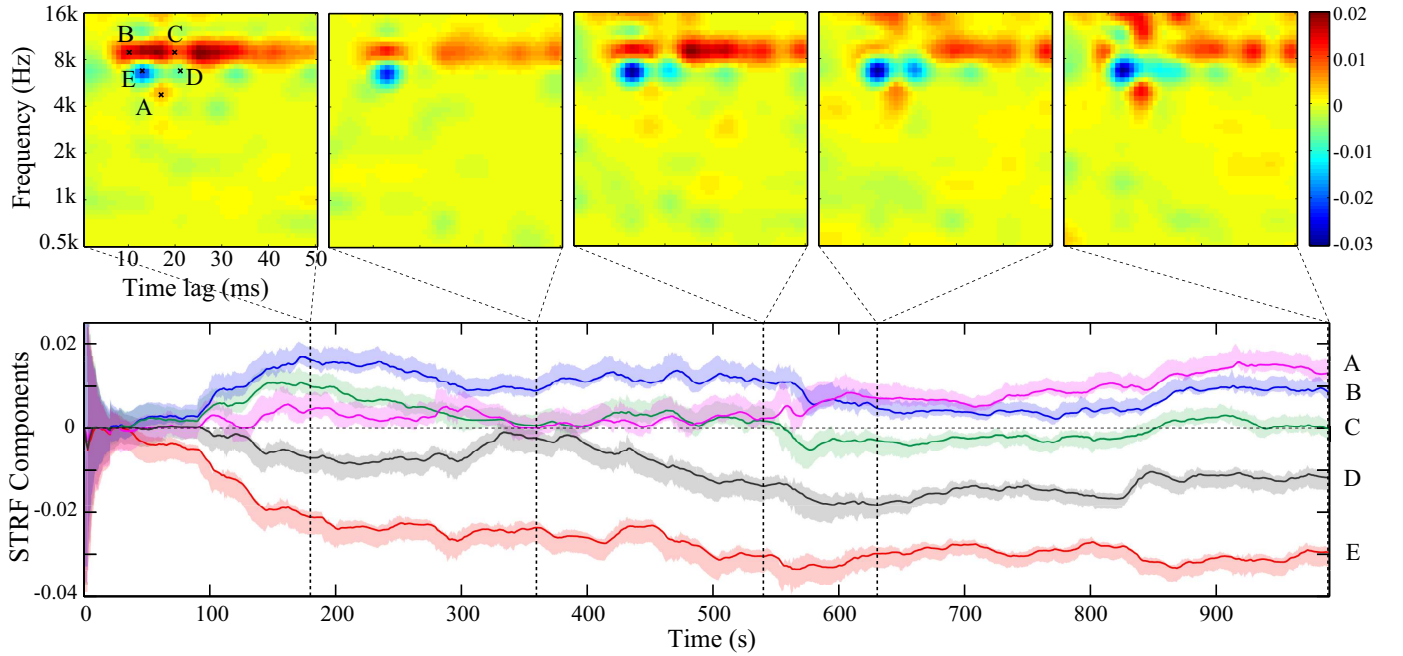


Fig. 4. The time-course of task-dependent STRF plasticity of a ferret A1 neuron. The top row shows snapshots of the STRF at five selected points in time, marked by the dashed vertical lines in the bottom graph. The bottom graph shows the time-course of five selected points (A through E) in the STRF marked on the leftmost panel of the top row.

We apply our proposed adaptive filter ℓ_1 -PPF₁ to multi-unit spike recordings from the ferrets A1 during a series of passive listening conditions and active auditory task conditions. During each active task, ferrets attended to the temporal dynamics of the sounds, and discriminated the rate of acoustic clicks [33]. The STRFs were estimated from the passive condition, where the quiescent animal listened to a series of broadband noise-like acoustic stimuli known as Temporally Orthogonal Ripple Combinations (TORC). The experiment consisted of 2 active and 11 passive blocks. Within each passive block, 30 TORCs were randomly repeated a total of 4-5 times each. In our analysis, we pool the spiking data corresponding to the same repeated TORC within each block. Therefore, the time axis corresponds to the experiment time modulo repetitions within each block. We discretize the resulting duration of $T = 990s$ to time bins of size $\Delta = 1 ms$, and segment data to windows of size $W = 10$ samples (10 ms). The STRF dimensions are 50×50 , regularly spanning lags of 1 to 50 ms and frequency bands of 0.5 kHz to 16 kHz (in logarithmic scale). The dictionary \mathbf{F} consists of 13×13 Gabor atoms, evenly spaced within the STRF domain. Each atom is a two-dimensional Gaussian kernel with a variance of $D^2/4$ per dimension, where D denotes the spacing between the atoms. We selected a forgetting factor of $\beta = 0.9998$, a step size of $\alpha = \frac{4(1-\beta)}{MW\bar{\sigma}^2}$, where $\bar{\sigma}^2$ is the average variance of the spectrogram components, $R = 1$ iterations per sample, and a regularization parameter of $\gamma = 40$ via two-fold even-odd cross validation.

Figure 4, top row, depicts five snapshots taken at $\{180, 360, 540, 630, 990\}$ sec corresponding to the end-points of the $\{2, 4, 6, 7, 11\}$ th passive tasks. The bottom row shows the time-course of five selected points (marked as A through D in the leftmost panel of the top row) of the STRF during the experiment. The STRF snapshots at times 180 and 540 sec

correspond to 90 secs after the two active tasks, respectively, and verify the sharpening effect of the excitatory region ($\sim 30 msec, 8 kHz$) due to the animal's attentive behavior following the active task reported in [22]. Moreover, the STRF snapshots at times 360 and 630 sec reveal the weakening of the excitatory region long after the active task and returning to the pre-active state, highlighting the plasticity of A1 neurons. Previous studies have revealed the STRF dynamics with a resolution of the order of minutes [34]. Our result in Figure 4 provides a temporal resolution of the order of centiseconds (3 orders of magnitude increase), while capturing the STRF sparsity in a robust fashion.

V. CONCLUDING REMARKS

In this paper, we considered recursive estimation of the time-varying parameter vectors in a logistic regression model for binary time series driven by continuous input. To this end, we integrated several techniques from compressed sensing, adaptive filtering, optimization and statistics. We constructed an objective function which enjoys from the trackability features of the RLS-type algorithms, sparsifying features of ℓ_1 -minimization, and unlike the rate-based linear models commonly used to analyze spiking data, takes into account the binary statistics of the observations. We analyzed the maximizers of the objective function in a rigorous fashion, revealing novel trade-offs between various model parameters. We constructed two adaptive filters, with respective linear and quadratic complexity requirements, for recursive maximization of the objective function in an online setting. Moreover, we characterized the statistical confidence regions for our estimates, and devised a recursive procedure to compute them efficiently.

Although we specialized our treatment to logistic statistics and ℓ_1 -regularization, our approach to algorithm development has a plug-and-play feature: other GLM link functions (e.g.,

log-link) with possibly history dependent covariates and other regularization schemes (e.g., re-weighted ℓ_1 , or group-sparse regularization) can be used instead and result in a large class of adaptive filters for sparse point process regression. We tested the performance of our algorithms on simulated as well as experimentally recorded spiking data. Our simulation studies revealed that the proposed filters outperform several existing point process filters. Application of our filters to real data from the ferret primary auditory cortex provided a high-resolution characterization of the time-course of spectrotemporal receptive field plasticity, with 3 orders of magnitudes increase in temporal resolution. Although we focused on auditory neurons, we expect a similar superior performance of our filters when applied to other sensory or motor neurons (e.g., cells in primary or supplementary motor cortex [35]).

APPENDIX A PROOF OF THEOREM 1

The proof is mainly based on the beautiful treatment of Negahban et al. [18]. The major difficulty in our case lies in the high inter-dependence of the covariates, which form a Toeplitz structure due to the setup of adaptive filtering. We address the latter issue by adopting techniques from another remarkable paper by Haupt et al. [26] to deal with the underlying interdependence. In the process, we also employ concentration inequalities for dependent random variables due to van de Geer [36].

In order to proceed, we adopt the notion of Strong Restricted Convexity (RSC) introduced in [18]. For a twice differentiable log-likelihood with respect to ω , the RSC property of order L implies the existence of a lower quadratic bound on the negative log-likelihood:

$$\mathcal{D}_{\mathcal{L}}(\Delta, \omega) := -\mathcal{L}^\beta(\omega + \Delta) + \mathcal{L}^\beta(\omega) + \Delta' \nabla \mathcal{L}^\beta(\omega) \geq \kappa \|\Delta\|_2^2,$$

for a positive constant $\kappa > 0$ and all $\Delta \in \mathbb{R}^M$ satisfying:

$$\|\Delta_{S^c}\|_1 \leq 3\|\Delta_S\|_1 + 4\sigma_S(\omega). \quad (27)$$

for any index set $S \subset \{1, 2, \dots, M\}$ of cardinality L . The following key lemma establishes the RSC for $\mathcal{L}^\beta(\omega)$:

Lemma 1: Let $\{\mathbf{x}_t\}_{t=1}^{KW}$ denote a sequence of covariates and let ω denote the corresponding logistic parameters. Then, for a fixed positive constant $d > 0$, there exist constants C' and $\kappa > 0$ such that for $\beta \geq 1 - \frac{C'}{L^2 \log M}$ and $K \geq \frac{\log 2}{\log(\frac{1}{\beta})}$ the negative log-likelihood $-\mathcal{L}^\beta(\omega)$ satisfies the RSC of order L with constant $\frac{\kappa}{1-\beta}$ with probability greater than $1 - \frac{3}{M^d}$. The constants C' and κ are only functions of d , p_{\min} , p_{\max} , σ^2 , B , W , and are explicitly given in the proof.

Proof: The proof is inspired by the elegant treatment of Negahban et al. [18]. The major difficulty in our setting is the high interdependence of successive covariates due to the shift structure induced by the adaptive setting, whereas in [18], the matrix of covariates is composed of i.i.d. rows. Using the Taylor's theorem, $\mathcal{D}_{\mathcal{L}}(\Delta, \omega)$ can be written as:

$$\sum_{i=1}^K \sum_{j=1}^W \beta^{K-i} \frac{\exp(\mathbf{x}'_{(i-1)W+j} \omega^*) |\Delta' \mathbf{x}_{(i-1)W+j}|^2}{(1 + \exp(\mathbf{x}'_{(i-1)W+j} \omega^*))^2}, \quad (28)$$

with $\omega^* = \omega + \tau \Delta$ for some $\tau \in (0, 1)$. Since by hypothesis $0 < p_{\min} \leq \lambda_i \Delta \leq p_{\max} < 1$, we have:

$$\frac{\exp(\mathbf{x}'_{(i-1)W+j} \omega^*)}{(1 + \exp(\mathbf{x}'_{(i-1)W+j} \omega^*))^2} \geq p_{\min}(1 - p_{\max}). \quad (29)$$

We can therefore further lower bound $\mathcal{D}_{\mathcal{L}}(\Delta, \omega)$ by:

$$\mathcal{D}_{\mathcal{L}}(\Delta, \omega) \geq p_{\min}(1 - p_{\max}) \sigma^2 N_\beta \{\Delta' \mathbf{C}_\beta \Delta\}, \quad (30)$$

where $N_\beta := W \frac{1-\beta^{K+1}}{1-\beta}$, and

$$\mathbf{C}_\beta := \frac{1}{\sigma^2 N_\beta} \sum_{i=1}^K \sum_{j=1}^W \beta^{K-i} \mathbf{x}_{(i-1)W+j} \mathbf{x}'_{(i-1)W+j}. \quad (31)$$

Note that the matrix \mathbf{C}_β has highly inter-dependent elements due to the Toeplitz structure in the adaptive design. In order to establish the RSC condition, we show the stronger Restricted Eigenvalue (RE) property, which in turn implies RSC [37]. Let $\delta \in (0, 1)$ be fixed. To do so, we need to bound the eigenvalues of $(\mathbf{C}_\beta)_S$, the restriction of \mathbf{C}_β to a subset of columns and rows corresponding to indices in $S \subset \{1, 2, \dots, M\}$ with $|S| = rL$, for some integer $r > 1 + \frac{9(1+\delta)}{1-\delta}$.

Without loss of generality, we assume that the first entry of the covariate vectors \mathbf{x}_t is replaced by σ instead of 1, for presentational simplicity of the following treatment. For $m, m' \neq 1$, we have:

$$(\mathbf{C}_\beta)_{m,m'} = \frac{1}{\sigma^2 N_\beta} \sum_{i=1}^K \sum_{j=0}^{W-1} \beta^{K-i} s_{(i-1)W+j+m-M} \times s_{(i-1)W+j+m'-M}.$$

For $m = m' = 1$,

$$(\mathbf{C}_\beta)_{1,1} = \frac{1}{\sigma^2 N_\beta} \sum_{i=1}^K W \beta^{K-i} \sigma^2 = \frac{1}{N_\beta} W \frac{1-\beta^{K+1}}{1-\beta} = 1,$$

and for $m \neq 1$,

$$(\mathbf{C}_\beta)_{m,1} = (\mathbf{C}_\beta)_{1,m} = \frac{1}{\sigma N_\beta} \sum_{i=1}^K \sum_{j=0}^{W-1} \beta^{K-i} s_{(i-1)W+j+m-M}.$$

We also have $\mathbb{E}\{(\mathbf{C}_\beta)_{m,m'}\} = \delta_{mm'}$. Using Hoeffding's inequality [38] we get:

$$\begin{aligned} \mathbb{P}(|(\mathbf{C}_\beta)_{m,m} - 1| > t) &\leq 2 \exp\left(-\frac{2t^2 N_\beta^2}{B^4 \sum_{i=1}^K W \beta^{2(K-i)}}\right) \\ &= 2 \exp\left(-\frac{2N_\beta^2 t^2 \sigma^4}{B^4 N_{\beta^2}}\right) \\ &\leq 2 \exp\left(-\frac{2N_\beta t^2 \sigma^4}{B^4}\right), \end{aligned} \quad (32)$$

since $N_{\beta^2} = N_\beta \frac{1+\beta^{K+1}}{1+\beta} \leq N_\beta$, for $\beta \in [0, 1]$. Similarly,

$$\mathbb{P}(|(\mathbf{C}_\beta)_{1,m}| > t) \leq 2 \exp\left(-\frac{2N_\beta t^2 \sigma^2}{B^2}\right),$$

By adopting the partitioning technique of Theorem 4 in [26], we also get for $m \neq m'$:

$$\mathbb{P}(|(\mathbf{C}_\beta)_{m,m'}| > t) \leq 4 \exp\left(-\frac{N_\beta t^2 \sigma^4}{8B^4}\right).$$

Let $B_0 := \max\{B^2, \frac{B^4}{\sigma^2}\}$. Now, using the union bound we have:

$$\mathbb{P}\left(\bigcup_{\substack{m, m'=1 \\ m \neq m'}}^M \left\{ |(\mathbf{C}_\beta)_{m, m'}| > \frac{\delta}{2rL} \right\}\right) \leq 2M^2 \exp\left(-\frac{N_\beta \delta^2 \sigma^2}{32B_0 r^2 L^2}\right),$$

where we have used $\binom{M}{2} < \frac{M^2}{2}$ and,

$$\mathbb{P}\left(\bigcup_{m=1}^M \left\{ |(\mathbf{C}_\beta)_{m, m} - 1| > \frac{\delta}{2rL} \right\}\right) \leq 2M \exp\left(-\frac{N_\beta \delta^2 \sigma^2}{4B_0 r^2 L^2}\right).$$

Now, by invoking the Gershgorin's disc theorem, the eigenvalues of any sub-matrix of \mathbf{C}_β restricted to an index set S with $|S| = rL$, lie in the interval $[1 - \delta, 1 + \delta]$ with probability at least:

$$\begin{aligned} 1 - 2M^2 \exp\left(-\frac{N_\beta \delta^2 \sigma^2}{32B_0 r^2 L^2}\right) - 2M \exp\left(-\frac{N_\beta \delta^2 \sigma^2}{4B_0 r^2 L^2}\right) \\ \geq 1 - 3M^2 \exp\left(-\frac{N_\beta \delta^2 \sigma^2}{32B_0 r^2 L^2}\right). \end{aligned} \quad (33)$$

Hence, by choosing $N_\beta \geq \frac{32B_0 \sigma^2 r^2 (d+2)}{\delta^2} L^2 \log M$, the probability above is greater than $1 - \frac{3}{M^d}$.

Next, by invoking Lemma 4.1 (ii) of [37], we have that \mathbf{C}_β satisfies the RSC condition over the set given by Eq. (27) with a constant given by:

$$\kappa_0 = (1 - \delta) \left(1 - 3\sqrt{\frac{1+\delta}{(r-1)(1-\delta)}}\right)^2. \quad (34)$$

Hence, the negative log-likelihood satisfies the RSC with a constant given by $p_{\min}(1 - p_{\max})\sigma^2 N_\beta \kappa_0$. Finally, by taking $K \geq \frac{\log 2}{\log(\frac{1}{\beta})}$, we have that $N_\beta \geq \frac{W}{2(1-\beta)}$, which makes κ independent of K and β , given by:

$$\kappa := \frac{p_{\min}(1 - p_{\max})\sigma^2 \kappa_0 W}{2}, \quad (35)$$

and $\beta \geq 1 - \frac{C'}{L^2 \log M}$ with $C' := \frac{W\delta^2}{64B_0 \sigma^2 r^2 (d+2)}$.

Next, the result of Theorem 1 of [18] implies:

$$\|\hat{\omega} - \omega\|_2 \leq \frac{2\gamma\sqrt{L}}{\kappa} + \sqrt{\frac{2\gamma\sigma_L(\omega)}{\kappa}}, \quad (36)$$

for $\gamma > 2\|\nabla \mathcal{L}^\beta(\omega)\|_\infty$. We have, for $m \neq 1$,

$$\begin{aligned} (\nabla \mathcal{L}^\beta(\omega))_m &= \sum_{i=1}^K \sum_{j=1}^W \beta^{K-i} s_{(i-1)W+j+m-M+1} \\ &\quad \{n_{(i-1)W+j+m-M+1} - \lambda_{(i-1)W+j+m-M+1} \Delta\}. \end{aligned}$$

Now, let \mathcal{F}_t be the σ -field generated by s_{-M+1}, \dots, s_t , i.e., $\sigma(s_{-M+1}, \dots, s_t)$. We have that

$$\begin{aligned} \mathbb{E}[\{n_t - \lambda_t \Delta\} s_t] &= \mathbb{E}[\mathbb{E}[\{n_t - \lambda_t \Delta\} s_t | \mathcal{F}_t]] \\ &= \mathbb{E}[s_t \mathbb{E}[\{n_t - \lambda_t \Delta\} | \mathcal{F}_t]] \\ &= \mathbb{E}\left[s_t \underbrace{\mathbb{E}[\{\lambda_t \Delta - \lambda_t \Delta\} | \mathcal{F}_t]}_{=0}\right] = 0. \end{aligned}$$

Hence for all m , $\mathbb{E}\{(\nabla \mathcal{L}^\beta(\omega))_m\} = 0$. We next invoke the following result for concentration of dependent random variables:

Proposition 1: Consider a sequence of σ -fields $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \dots$. Suppose that X_i is \mathcal{F}_i -measurable with $|X_i| \leq B_i$ for some constant B_i , $i = 1, 2, \dots$ and $\mathbb{E}\{X_i | \mathcal{F}_{i-1}\} = 0$. Then for all $t > 0$,

$$\mathbb{P}\left(\sum_{i=1}^n X_i \geq t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n B_i^2}\right).$$

Proof: This result is a special case of Theorem 2.5 of [36] for *bounded* and possibly dependent random variables, which generalizes Hoeffding's inequality. ■

In our case, we can similarly show that $\mathbb{E}[s_t \{n_t - \lambda_t \Delta\} | \mathcal{F}_{t-1}] = \mathbb{E}[s_t \mathbb{E}[\{n_t - \lambda_t \Delta\} | \mathcal{F}_{t-1}, \mathcal{F}_t]] = 0$. Moreover, each summand is bounded by $2\beta^{K-i}B$. Hence, using the result of Proposition 1, by taking $n = TW$ and $X_i = s_i[n_i - \lambda_i \Delta]$, we get:

$$\begin{aligned} \mathbb{P}(|(\nabla \mathcal{L}^\beta(\omega))_m| > tN_\beta) &\leq 2 \exp\left(-\frac{t^2 N_\beta^2}{2 \sum_{i=1}^K W \beta^{2(K-i)}}\right) \\ &\leq 2 \exp\left(-\frac{N_\beta t^2}{2}\right). \end{aligned}$$

Using the union bound, we have:

$$\mathbb{P}(\|\nabla \mathcal{L}^\beta(\omega)\|_\infty > tN_\beta) \leq 2M \exp\left(-\frac{N_\beta t^2}{2}\right). \quad (37)$$

By choosing $t = \sqrt{\frac{2(d+1)\log M}{N_\beta}}$, we have that

$$\|\nabla \mathcal{L}^\beta(\omega)\|_\infty < \sqrt{2(d+1)N_\beta \log M}, \quad (38)$$

with probability at least $1 - \frac{2}{M^d}$.

Hence, for a fixed $\delta < 1$, $d > 0$, and $r > 1 + \frac{9(1+\delta)}{1-\delta}$, by taking $\beta \geq 1 - \frac{C'}{L^2 \log M}$ with $C' := \frac{W\delta^2}{64B_0 \sigma^2 r^2 (d+2)}$ and $\gamma > C'' \sqrt{\frac{\log M}{1-\beta}}$ with $C'' := \sqrt{4(d+1)W}$, any maximizer $\hat{\omega}$ satisfies:

■ $\|\hat{\omega} - \omega\|_2 \leq C\sqrt{(1-\beta)L \log M} + \sqrt{C\sigma_L(\omega)} \sqrt[4]{(1-\beta)L \log M}$, with probability at least $1 - \frac{3}{M^d} - \frac{2}{M^d}$, where C is given by

$$C := \frac{\sqrt{128(d+1)}}{\sqrt{W} p_{\min}(1 - p_{\max})\sigma^2(1 - \delta) \left(1 - 3\sqrt{\frac{1+\delta}{(r-1)(1-\delta)}}\right)^2}.$$

APPENDIX B

THE PROXIMAL GRADIENT ALGORITHM

In this appendix, we give an overview of the proximal gradient algorithm for minimization of convex function. The corresponding algorithm for maximization of concave functions can be obtained by negating the objective functions. Consider the general optimization problem

$$\min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{x}), \quad (39)$$

where functions $f(\mathbf{x}) : \mathbb{R}^M \rightarrow \mathbb{R}$ and $g(\mathbf{x}) : \mathbb{R}^M \rightarrow \mathbb{R} \cup \{\infty\}$ are assumed to be closed proper convex functions. Suppose that f is differentiable with a Lipschitz continuous gradient ∇f with constant $L(\nabla f)$. The function g can be possibly non-smooth. A wide range of practical optimization problems can be cast in this form, particularly in the context of machine

learning [39], where the objective function can be decomposed into a loss function and a regularization term.

The proximal gradient algorithm provides an iterative procedure for solving (39) in the following form:

$$\mathbf{x}^{(\ell+1)} = \mathcal{P}_{\alpha^{(\ell)}g} \left[\mathbf{x}^{(\ell)} - \alpha^{(\ell)} \nabla f(\mathbf{x}^{(\ell)}) \right], \quad (40)$$

where the parameter $\alpha^{(\ell)}$ is an appropriately chosen step size at iteration ℓ so that $\alpha^{(\ell)} < \frac{1}{L(\nabla f)}$, and the *proximal operator* $\mathcal{P}_{\alpha g}(\cdot)$ of function g with parameter α is defined as

$$\mathcal{P}_{\alpha g}(\mathbf{x}) := \underset{\mathbf{u}}{\operatorname{argmin}} \left\{ g(\mathbf{u}) + \frac{1}{2\alpha} \|\mathbf{u} - \mathbf{x}\|_2^2 \right\}. \quad (41)$$

Among the several interpretations available for the proximal gradient method, we have adopt a quadratic approximation-based model to derive the main iterative scheme in (40). This interpretation [40], [41], is based on the Majorization-Minimization algorithm (see [42] for a detailed discussion). In the approximation-based derivation, the ℓ -th iteration for solving the general problem (39) can be written in the following form:

$$\mathbf{x}^{(\ell+1)} = \underset{\mathbf{u}}{\operatorname{argmin}} \left\{ \hat{f}_\alpha(\mathbf{x}, \mathbf{x}^{(\ell)}) + g(\mathbf{x}), \right. \quad (42)$$

where the original objective function f is replaced with a quadratically-regularized linear approximation around the previous iterate $\mathbf{x}^{(\ell)}$, given by

$$\hat{f}_\alpha(\mathbf{x}, \mathbf{y}) := f(\mathbf{y}) + \nabla f(\mathbf{y})'(\mathbf{x} - \mathbf{y}) + \frac{1}{2\alpha} \|\mathbf{x} - \mathbf{y}\|_2^2, \quad (43)$$

where the quadratic term is referred to as the *trust region penalty*. Modulo constants, the objective function in (42) can be rearranged to get the proximal gradient form

$$\begin{aligned} \mathbf{x}^{(\ell+1)} &= \underset{\mathbf{x}}{\operatorname{argmin}} \left\{ g(\mathbf{x}) + \frac{1}{2\alpha} \|\mathbf{x} - \mathbf{x}^{(\ell)} + \alpha \nabla f(\mathbf{x}^{(\ell)})\|_2^2 \right\} \\ &= \mathcal{P}_{\alpha g} \left[\mathbf{x}^{(\ell)} - \alpha \nabla f(\mathbf{x}^{(\ell)}) \right]. \end{aligned} \quad (44)$$

The proximal operator often admits closed form expressions. As for ℓ_1 -regularization, the proximal operator takes the simple form of the *soft thresholding shrinkage operator* $\mathcal{P}_{\alpha \|\cdot\|_1} := \mathcal{S}_\alpha$ whose i th component is given by

$$(\mathcal{S}_\alpha(x))_i := \operatorname{sgn}(x_i)(x_i - \alpha)_+, \quad (45)$$

with sgn denoting the standard signum function, and $(a)_+ := \max\{a, 0\}$. In this case, the proximal algorithm leads to a family of algorithms referred to as iterative shrinkage algorithms [43]–[45], where each iteration involves a simple gradient descent step followed by a shrinkage operation.

Finally, in our setting, the function f is taken to be the exponentially weighted log-likelihood $\mathcal{L}^\beta(\cdot)$. Due to the smoothness of the logistic function, the Lipschitz constant for $\nabla \mathcal{L}^\beta(\omega_k)$ can be upper bounded by the trace of the Hessian $\mathbf{B}_k(\omega_k)$ given in Eq. (21). Noting that the elements of $\mathbf{\Lambda}_i$ are at most equal to $1/4$, we get $L(\nabla \mathcal{L}^\beta(\omega_k)) \leq \frac{1}{4} \sum_{i=1}^k \sum_{j=1}^W \beta^{k-i} x_{(i-1)W+j}^2$. Using assumption (1) of Section III-B and an application of Hoeffding's inequality, shows that the sum is concentrated around its mean given by $\frac{MW\sigma^2}{4(1-\beta)}$, for large enough k . Therefore, we choose the step size $\alpha = \frac{(1-\beta)}{cMW\sigma^2}$, for some constant $c \geq 1/4$.

APPENDIX C

COMPUTATION OF CONFIDENCE INTERVALS

The ℓ_1 -regularized ML estimate of Eq. (11) can be written in the following form

$$\hat{\omega}_k = \underset{\omega_k}{\operatorname{argmax}} \{ \mathfrak{P}_\beta \mathcal{L}(\omega_k) - \gamma \|\omega_k\|_1 \}, \quad (46)$$

where $\mathcal{L}(\omega) := \log p(\mathbf{n}|\mathbf{X}, \omega)$ denotes the log-likelihood function over a generic window with spiking vector \mathbf{n} , data matrix \mathbf{X} and parameter vector ω , and the operator $\mathfrak{P}_\beta f(\mathbf{n}, \mathbf{X}, \omega)$ is defined for a function $f : \{0, 1\}^W \times \mathbb{R}^{W \times M} \times \mathbb{R}^M \rightarrow \mathbb{R}$ as the empirical expectation exponentially weighted with a forgetting factor β :

$$\mathfrak{P}_\beta f(\omega_k) = \sum_{i=1}^k \beta^{k-i} f(\mathbf{n}_i, \mathbf{X}_i; \omega_k), \quad (47)$$

where we have suppressed the dependence of f on \mathbf{n} and \mathbf{X} on the left hand side for notational simplicity. Following the treatment of Theorem 3.1 of [36], the corresponding empirical gradient vector and Hessian are respectively given by:

$$\mathbf{g}_k(\omega_k) := \mathfrak{P}_\beta \nabla \mathcal{L}(\omega_k) = \sum_{i=1}^k \beta^{k-i} \mathbf{X}_i' \boldsymbol{\varepsilon}_i(\omega_k), \quad (48)$$

$$\mathbf{B}_k(\omega_k) := \mathfrak{P}_\beta \nabla^2 \mathcal{L}(\omega_k) = - \sum_{i=1}^k \beta^{k-i} \mathbf{X}_i' \mathbf{\Lambda}_i(\omega_k) \mathbf{X}_i. \quad (49)$$

The KKT conditions for the estimator $\hat{\omega}_k$ can be then written as:

$$\mathbf{g}_k(\hat{\omega}_k) - \gamma \hat{\mathbf{s}}_k = 0, \quad \|\hat{\mathbf{s}}_k\|_\infty \leq 1. \quad (50)$$

where $\hat{\mathbf{s}}_k \in \partial \|\hat{\omega}_k\|_1$ is a subgradient vector from the subdifferential of the ℓ_1 norm, with components $(\hat{\mathbf{s}}_k)_m = \operatorname{sgn}((\hat{\omega}_k)_m)$ for $(\hat{\omega}_k)_m \neq 0$ and $|(\hat{\mathbf{s}}_k)_m| \leq 1$ otherwise, for $m = 1, 2, \dots, M$. Substituting $\mathfrak{P}_\beta \mathcal{L}(\omega_k)$ by its quadratic approximation around the true parameter vector ω_k , and inverting the corresponding KKT conditions, the 'de-sparsified' estimator $\hat{\mathbf{w}}_k$ can be obtained as:

$$\hat{\mathbf{w}}_k := \hat{\omega}_k - \hat{\boldsymbol{\Theta}}_k \mathbf{g}_k(\hat{\omega}_k), \quad (51)$$

where the matrix $\hat{\boldsymbol{\Theta}}_k$ is the approximate inverse of Hessian matrix $\mathbf{B}_k(\hat{\omega}_k)$, and can be computed using the following node wise regression procedure [36]. To compute the m th row of $\hat{\boldsymbol{\Theta}}_k$, first the solution to the following LASSO problem is obtained:

$$\hat{\boldsymbol{\psi}}_m := \underset{\boldsymbol{\psi} \in \mathbb{R}^{M-1}}{\operatorname{argmin}} \left(-2(\mathbf{B}_k)_{m, \setminus m} \boldsymbol{\psi} + \boldsymbol{\psi}' (\mathbf{B}_k)_{\setminus m, \setminus m} \boldsymbol{\psi} + 2\gamma_m \|\boldsymbol{\psi}\|_1 \right),$$

where the dependence of \mathbf{B}_k on $\hat{\omega}_k$ is suppressed for notational convenience, and the subscript notations are the same as those described in the footnote of Algorithm 3. Then, we define the vector $\mathbf{c} \in \mathbb{R}^M$ as:

$$(\mathbf{c})_m = 1, \quad (\mathbf{c})_{\setminus m} = -\hat{\boldsymbol{\psi}}_m, \quad (52)$$

and the scaling constant τ_m^2 as

$$\tau_m^2 := (\mathbf{B}_k)_{m, m} - \hat{\boldsymbol{\psi}}_m^{(L)} (\mathbf{B}_k)_{\setminus m, \setminus m}' \quad (53)$$

Finally, the m th row of $\hat{\boldsymbol{\Theta}}_k$ is given by $(\hat{\boldsymbol{\Theta}}_k)_m := \frac{1}{\tau_m^2} \mathbf{c}$. The variance and the confidence interval at a level of α for the

m th component of $\hat{\omega}_k$ can then be computed as given in lines 9 and the output of Algorithm 3 [36], where

$$\mathbf{G}_k(\omega) := \mathfrak{P}_{\beta^2} \nabla \mathcal{L}(\omega) \nabla \mathcal{L}'(\omega) = \sum_{i=1}^k \beta^{2(k-i)} \mathbf{X}_i' \varepsilon_i(\omega) \varepsilon_i(\omega)' \mathbf{X}_i.$$

Using Taylor expansion similar to that in the development of ℓ_1 -PPF₁, the matrix $\mathbf{G}_k(\hat{\omega}_k)$ can be recursively updated as given in line 2 of Algorithm 3. Finally, the node wise regression can be recursively computed using the SPARLS algorithm [19], which is given in lines 3–5 of Algorithm 3. The parameter γ_m can be chosen to be in the same order of γ in (11).

REFERENCES

- [1] L. M. Frank, G. B. Stanley, and E. N. Brown, "Hippocampal plasticity across multiple days of exposure to novel environments," *The Journal of neuroscience*, vol. 24, no. 35, pp. 7681–7689, 2004.
- [2] D. A. Depireux, J. Z. Simon, D. J. Klein, and S. A. Shamma, "Spectrotemporal response field characterization with dynamic ripples in ferret primary auditory cortex," *Journal of neurophysiology*, vol. 85, no. 3, pp. 1220–1234, 2001.
- [3] D. J. Daley and D. Vere-Jones, *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007, vol. 2.
- [4] Y. Ogata, "Statistical models for earthquake occurrences and residual analysis for point processes," *Journal of the American Statistical Association*, vol. 83, no. 401, pp. 9–27, 1988.
- [5] D. Vere-Jones, "Stochastic models for earthquake occurrence," *Journal of the Royal Statistical Society. Series B*, pp. 1–62, 1970.
- [6] E. N. Brown, R. E. Kass, and P. P. Mitra, "Multiple neural spike train data analysis: state-of-the-art and future challenges," *Nature neuroscience*, vol. 7, no. 5, pp. 456–461, 2004.
- [7] E. N. Brown, D. P. Nguyen, L. M. Frank, M. A. Wilson, and V. Solo, "An analysis of neural receptive field plasticity by point process adaptive filtering," *Proceedings of the National Academy of Sciences*, vol. 98, no. 21, pp. 12 261–12 266, 2001.
- [8] A. Smith and E. N. Brown, "Estimating a state-space model from point process observations," *Neural Comp.*, vol. 15, no. 5, pp. 965–991, 2003.
- [9] L. Paninski, "Maximum likelihood estimation of cascade point-process neural encoding models," *Network: Comp. in Neural Systems*, vol. 15, no. 4, pp. 243–262, 2004.
- [10] L. Paninski, J. Pillow, and J. Lewi, "Statistical models for neural encoding, decoding, and optimal stimulus design," *Progress in brain research*, vol. 165, pp. 493–507, 2007.
- [11] J. W. Pillow, Y. Ahmadian, and L. Paninski, "Model-based decoding, information estimation, and change-point detection techniques for multi-neuron spike trains," *Neural Comp.*, vol. 23, no. 1, pp. 1–45, 2011.
- [12] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown, "A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects," *Journal of neurophysiology*, vol. 93, no. 2, pp. 1074–1089, 2005.
- [13] S. S. Haykin, *Adaptive filter theory*. Pearson Education India, 2008.
- [14] U. T. Eden, L. M. Frank, R. Barbieri, V. Solo, and E. N. Brown, "Dynamic analysis of neural encoding by point process adaptive filtering," *Neural comp.*, vol. 16, no. 5, pp. 971–998, 2004.
- [15] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Trans. on*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [16] E. J. Candès et al., "Compressive sampling," in *Proceedings of the International Congress of Mathematicians*, vol. 3. Madrid, Spain, 2006, pp. 1433–1452.
- [17] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.
- [18] S. N. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu, "A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers," *Statistical Science*, vol. 27, no. 4, pp. 538–557, 2012.
- [19] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The sparse RLS algorithm," *Signal Processing, IEEE Trans. on*, vol. 58, no. 8, pp. 4013–4025, 2010.
- [20] N. Kalouptsidis, G. Mileounis, B. Babadi, and V. Tarokh, "Adaptive algorithms for sparse system identification," *Signal Processing*, vol. 91, no. 8, pp. 1910–1919, 2011.
- [21] B. Dumitrescu, A. Onose, P. Helin, and I. Tábuş, "Greedy sparse RLS," *Signal Processing, IEEE Trans. on*, vol. 60, no. 5, pp. 2194–2207, 2012.
- [22] J. Fritz, S. Shamma, M. Elhilali, and D. Klein, "Rapid task-related plasticity of spectrotemporal receptive fields in primary auditory cortex," *Nature neuroscience*, vol. 6, no. 11, pp. 1216–1223, 2003.
- [23] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown, "A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects," *Journal of neurophysiology*, vol. 93, no. 2, pp. 1074–1089, 2005.
- [24] Z. Chen, D. F. Putrino, S. Ghosh, R. Barbieri, and E. N. Brown, "Statistical inference for assessing functional connectivity of neuronal ensembles with sparse spiking data," *Neural Systems and Rehabilitation Engineering, IEEE Trans. on*, vol. 19, no. 2, pp. 121–135, 2011.
- [25] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
- [26] J. Haupt, W. U. Bajwa, G. Raz, and R. Nowak, "Toeplitz compressed sensing matrices with applications to sparse channel estimation," *Information Theory, IEEE Trans. on*, vol. 56, no. 11, pp. 5862–5875, 2010.
- [27] A. Javanmard and A. Montanari, "Confidence intervals and hypothesis testing for high-dimensional regression," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2869–2909, 2014.
- [28] S. Van de Geer, P. Bühlmann, Y. Ritov, and R. Dezeure, "On asymptotically optimal confidence regions and tests for high-dimensional models," *The Annals of Stat.*, vol. 42, no. 3, pp. 1166–1202, 2014.
- [29] C.-H. Zhang and S. S. Zhang, "Confidence intervals for low dimensional parameters in high dimensional linear models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 76, no. 1, pp. 217–242, 2014.
- [30] N. Meinshausen and P. Bühlmann, "High-dimensional graphs and variable selection with the lasso," *The Annals of Stat.*, pp. 1436–1462, 2006.
- [31] E. N. Brown, R. Barbieri, V. Ventura, R. E. Kass, and L. M. Frank, "The time-rescaling theorem and its application to neural spike train data analysis," *Neural comp.*, vol. 14, no. 2, pp. 325–346, 2002.
- [32] R. Haslinger, G. Pipa, and E. Brown, "Discrete time rescaling theorem: determining goodness of fit for discrete time statistical models of neural spiking," *Neural comp.*, vol. 22, no. 10, pp. 2477–2506, 2010.
- [33] J. Fritz, M. Elhilali, and S. Shamma, "Active listening: task-dependent plasticity of spectrotemporal receptive fields in primary auditory cortex," *Hearing research*, vol. 206, no. 1, pp. 159–176, 2005.
- [34] N. Mesgarani, J. Fritz, and S. Shamma, "A computational model of rapid task-related plasticity of auditory cortical receptive fields," *Journal of computational neuroscience*, vol. 28, no. 1, pp. 19–27, 2010.
- [35] F. Gandolfo, C.-S. Li, B. Benda, C. P. Schioppa, and E. Bizzi, "Cortical correlates of learning in monkeys adapting to a new dynamical environment," *Proceedings of the National Academy of Sciences*, vol. 97, no. 5, pp. 2259–2263, 2000.
- [36] S. A. van de Geer, "On Hoeffding's inequality for dependent random variables," in *Empirical Process Techniques for Dependent Data*, H. Dehling and W. Philipp, Eds. Springer, 2001.
- [37] P. J. Bickel, Y. Ritov, and A. B. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *The Annals of Stat.*, pp. 1705–1732, 2009.
- [38] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American statistical association*, vol. 58, no. 301, pp. 13–30, 1963.
- [39] B. Scholkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2001.
- [40] M. A. Figueiredo and R. D. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *Image Processing, 2005. ICIP 2005. IEEE International Conf. on*, vol. 2. IEEE, 2005, pp. II–782.
- [41] M. A. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *Image Processing, IEEE Trans. on*, vol. 16, no. 12, pp. 2980–2991, 2007.
- [42] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, 2004.
- [43] M. A. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *Image Processing, IEEE Trans. on*, vol. 12, no. 8, pp. 906–916, 2003.
- [44] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM review*, vol. 51, no. 1, pp. 34–81, 2009.
- [45] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.